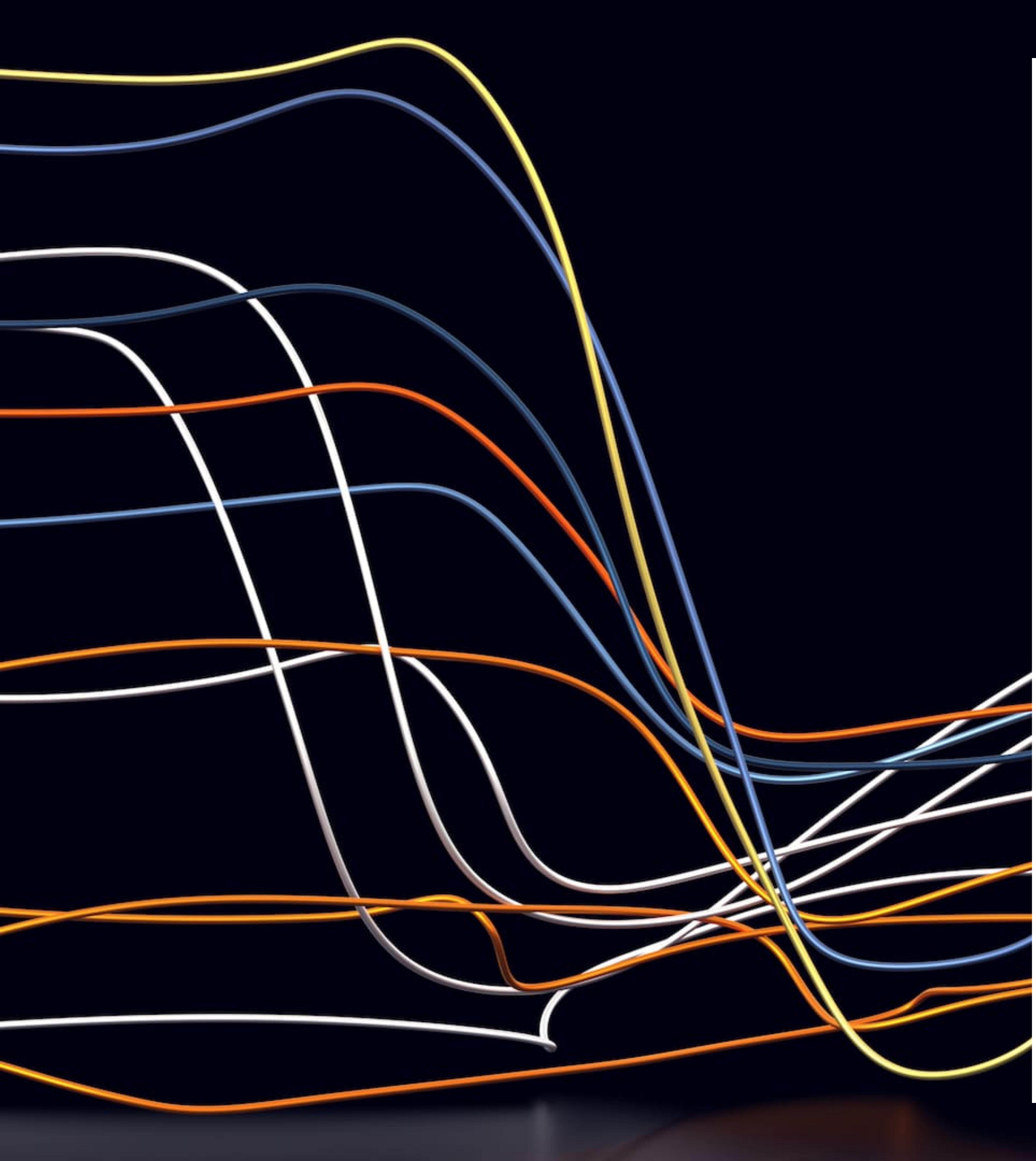


INNOQ TECHNOLOGY DAY / 13.11.2023

Data Contracts

OpenAPI for Data?

STEFAN NEGELE
CONSULTANT @ INNOQ

Data contracts are a tool for communicating and building shared expectations and understanding of data

Green Garden IoT Corp

CTO Team

Team Sensors

Procurement Department

Problems with data comprehension

- Field and table names are not sufficient to understand the content
- Field and model descriptions are not supported by all schema description technologies.
- Documentation for production code is often located elsewhere
- Explanations for field or model descriptions require further context

This leads to:

- Misinterpreted data
- Incorrect models
- Incorrect conclusions

Technical problems

Use of non-consensual APIs can break important analytical systems, e.g.:

- Direct Queries
- ETL pipelines
- Data products
- On-read schemas in data lakes

This leads to:

- Data reaches its target too late
- Unhappy Data Engineers

A clear interface definition that data consumers can rely on, and data producers can build upon.

Data Contract

The most important characteristics:

- Clear schema
- Clear semantics
- Clear guarantees on data quality
- Clear guarantees on availability
- Clear ownership

Cool, but what can I do with it?

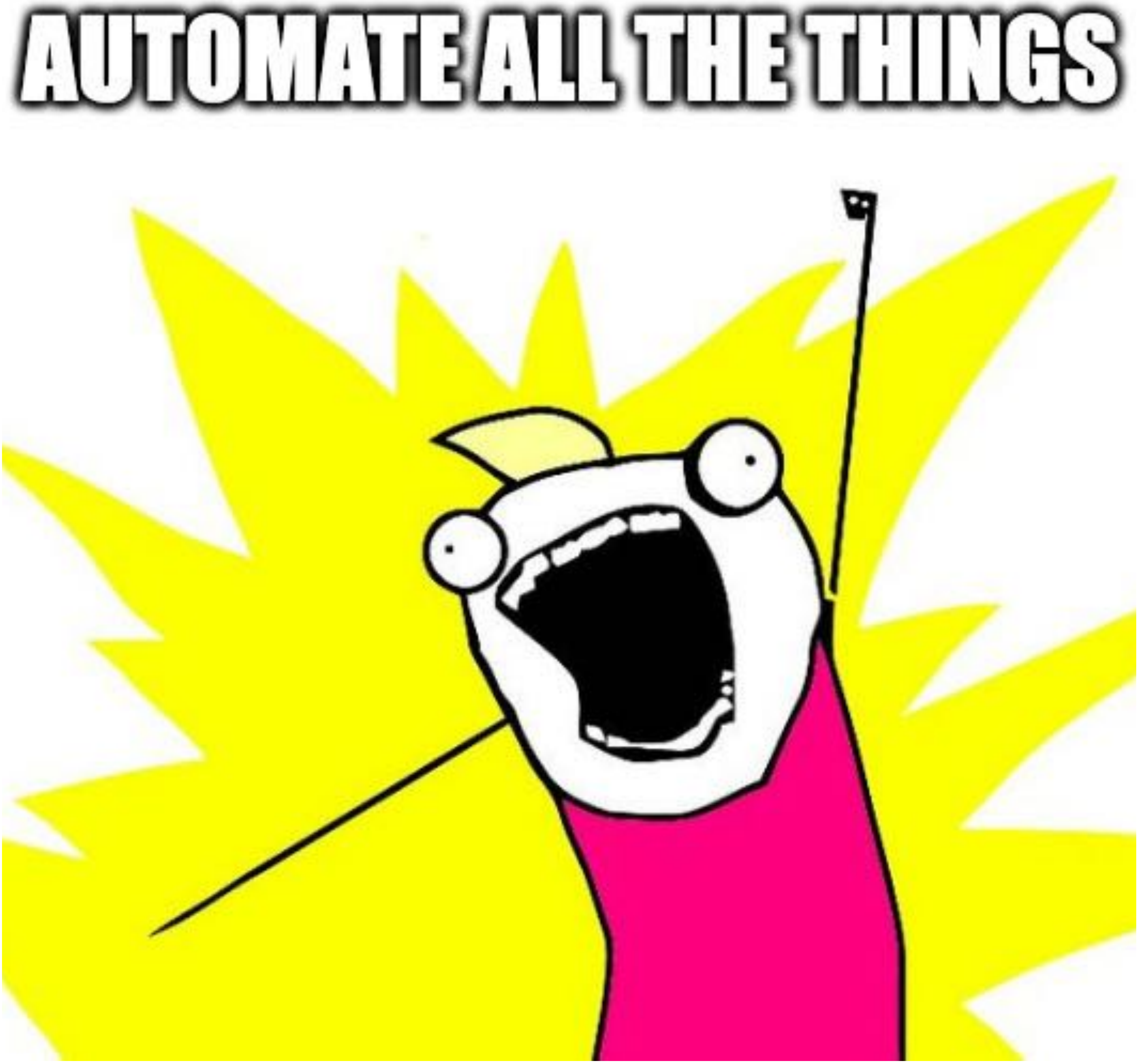
Basis for discussion

Bring together data providers and data consumers.

Talk about what they can provide and what is needed.

Automation

- Documentation
- Access control
- Resource provisioning
- Alerting
- Continuous integration



So, what do I need to put in the document?

Content characteristics

- Data Model (Schema & Semantics)
- Data quality
- How to access the data
- Ownership information
- Service level objectives
- Terms of service
- Sample data

Structural characteristics

- Machine readable
- Human readable
- Versionable
- Room for expansion
- Technology agnostic

I want to implement that, but how?

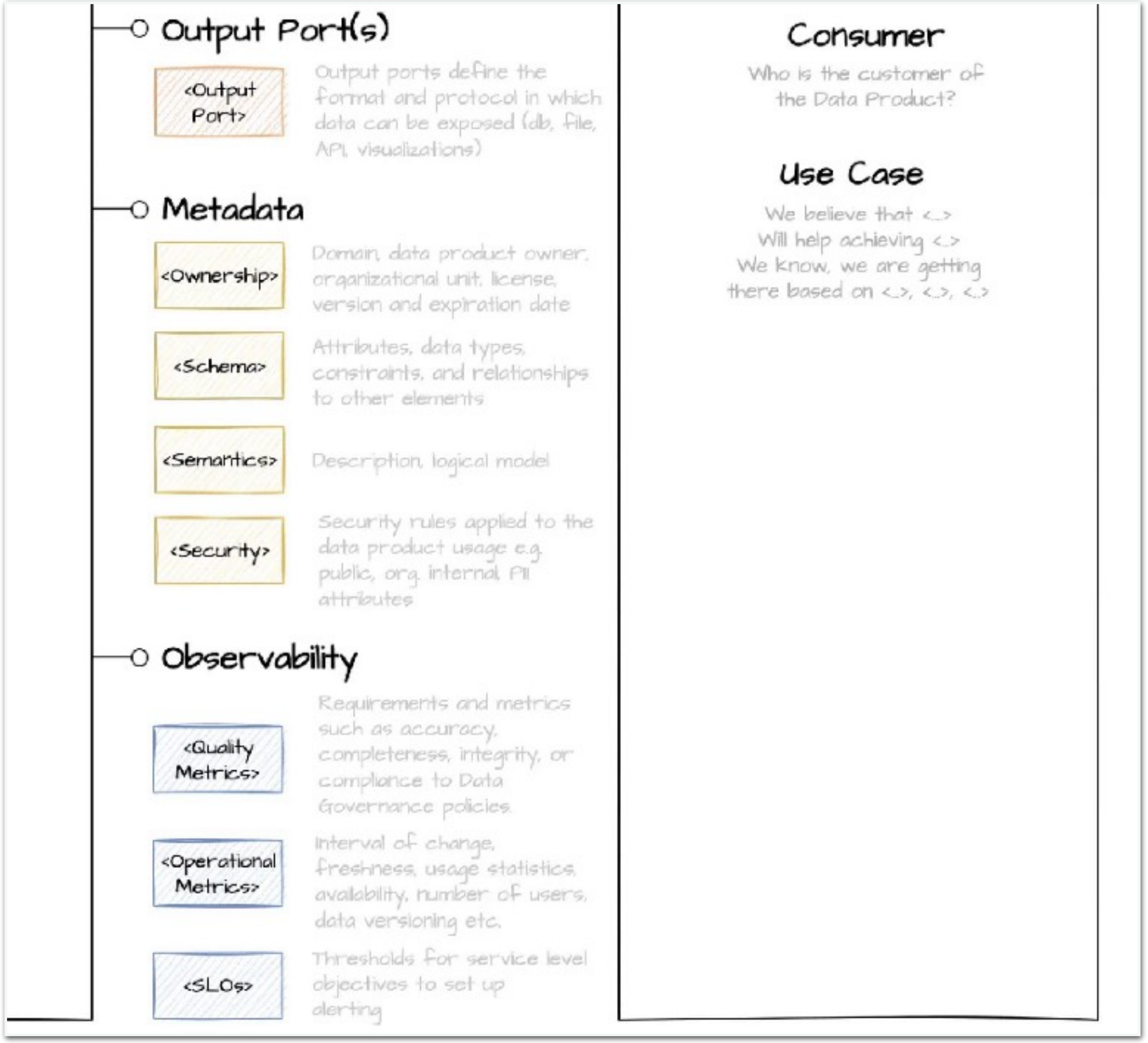
Discussion formats

- Provider driven
- Consumer driven
- Data contract workshop
- Change requests

Data Contract Workshop

Things to consider:

- Who takes part?
- Work with sticky notes or technical document?
- What has to be discussed?



Output Port(s)

<Output Port> Output ports define the format and protocol in which data can be exposed (db, file, API, visualizations)

Metadata

<Ownership> Domain, data product owner, organizational unit, license, version and expiration date

<Schema> Attributes, data types, constraints, and relationships to other elements

<Semantics> Description, logical model

<Security> Security rules applied to the data product usage e.g. public, org. internal, PII attributes

Observability

<Quality Metrics> Requirements and metrics such as accuracy, completeness, integrity, or compliance to Data Governance policies

<Operational Metrics> Interval of change, freshness, usage statistics, availability, number of users, data versioning etc.

<SLOs> Thresholds for service level objectives to set up alerting

Consumer

Who is the customer of the Data Product?

Use Case

We believe that <..>
Will help achieving <..>
We know, we are getting there based on <..>, <..>, <..>

datamesh-architecture.com/data-product-canvas

Now let's open Word, okay?

(No, we do not.)

JSON Schema

Structure

- 🟢 Machine readable
- 🟠 Human readable
- 🟠 Versionable
- 🟢 Room for expansion
- 🟢 Technology agnostic

Content

- 🟢 Data Model (Schema & Semantics)
- 🔴 Data quality
- 🔴 How to access the data
- 🔴 Ownership information
- 🔴 Service level objectives
- 🔴 Terms of service
- 🔴 Sample data

OpenAPI

Structure

- 🟢 Machine readable
- 🟠 Human readable
- 🟢 Versionable
- 🟢 Room for expansion
- 🟢 Technology agnostic

Content

- 🟢 Data Model (Schema & Semantics)
- 🔴 Data quality
- 🟢 How to access the data
- 🟢 Ownership information
- 🔴 Service level objectives
- 🟠 Terms of service
- 🟢 Sample data

Open Data Contract Standard

Structure

- 🟢 Machine readable
- 🟠 Human readable
- 🟢 Versionable
- 🟢 Room for expansion
- 🟠 Technology agnostic

Content

- 🟢 Data Model (Schema & Semantics)
- 🟢 Data quality
- 🟢 How to access the data
- 🟢 Ownership information
- 🟢 Service level objectives
- 🟢 Terms of service
- 🟢 Sample data

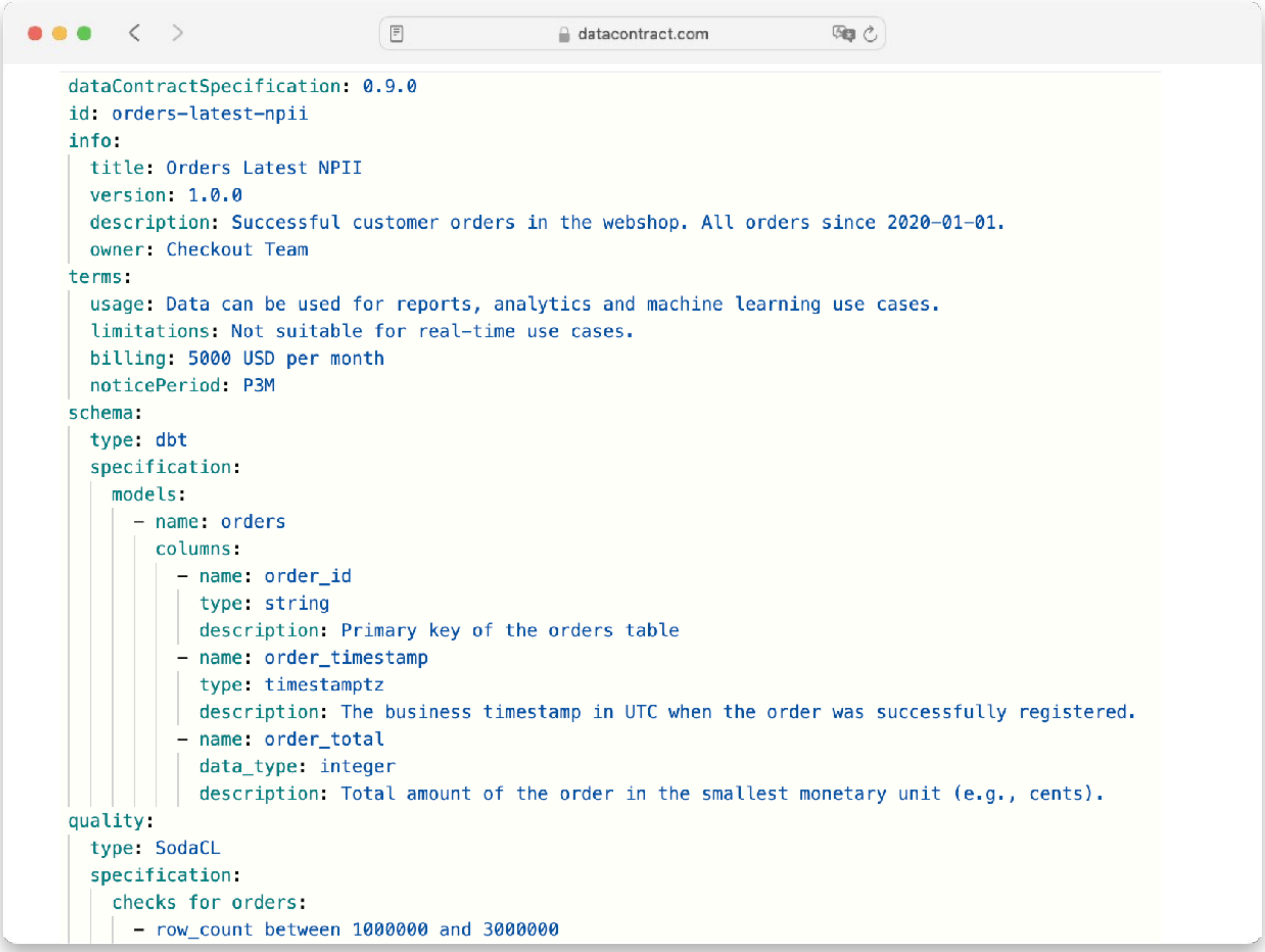
Data Contract Specification

Structure

- 🟢 Machine readable
- 🟢 Human readable
- 🟢 Versionable
- 🟢 Room for expansion
- 🟢 Technology agnostic

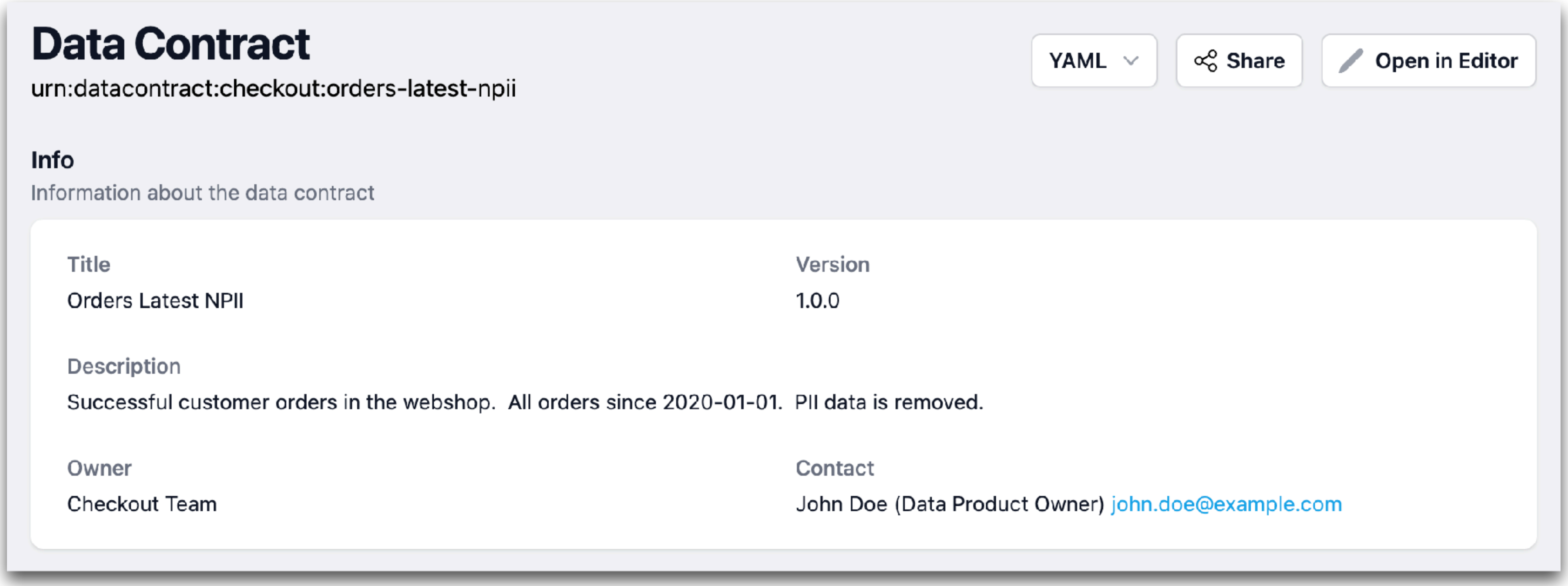
Content

- 🟢 Data Model (Schema & Semantics)
- 🟢 Data quality
- 🟢 How to access the data
- 🟢 Ownership information
- 🟢 Service level objectives
- 🟢 Terms of service
- 🟢 Sample data



```
dataContractSpecification: 0.9.0
id: orders-latest-npii
info:
  title: Orders Latest NPII
  version: 1.0.0
  description: Successful customer orders in the webshop. All orders since 2020-01-01.
  owner: Checkout Team
terms:
  usage: Data can be used for reports, analytics and machine learning use cases.
  limitations: Not suitable for real-time use cases.
  billing: 5000 USD per month
  noticePeriod: P3M
schema:
  type: dbt
  specification:
    models:
      - name: orders
        columns:
          - name: order_id
            type: string
            description: Primary key of the orders table
          - name: order_timestamp
            type: timestamptz
            description: The business timestamp in UTC when the order was successfully registered.
          - name: order_total
            data_type: integer
            description: Total amount of the order in the smallest monetary unit (e.g., cents).
quality:
  type: SodaCL
  specification:
    checks for orders:
      - row_count between 1000000 and 3000000
```

datacontract.com



Data Contract

urn:datacontract:checkout:orders-latest-npii

Info

Information about the data contract

Title

Orders Latest NPII

Version

1.0.0

Description

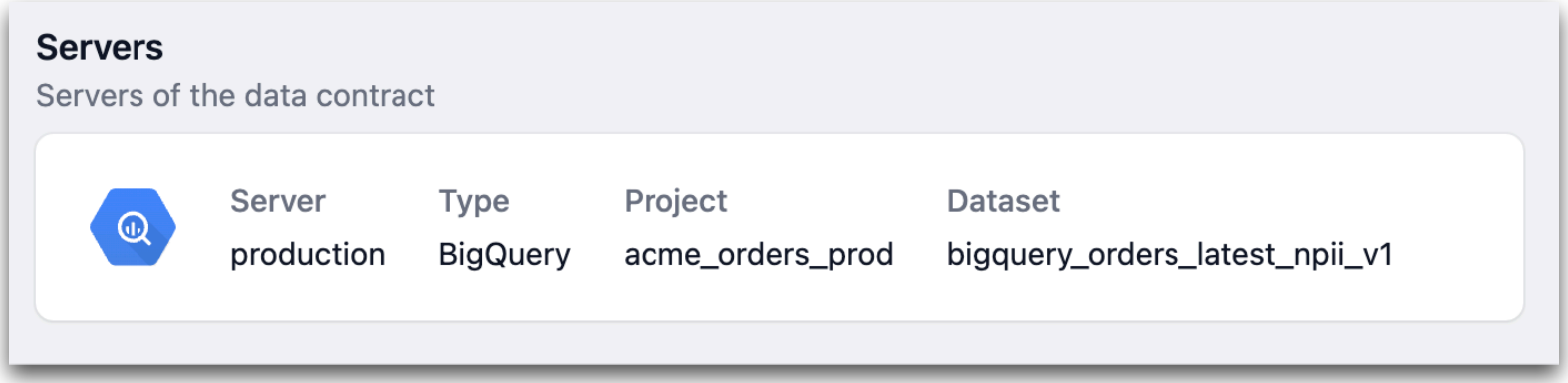
Successful customer orders in the webshop. All orders since 2020-01-01. PII data is removed.

Owner

Checkout Team

Contact

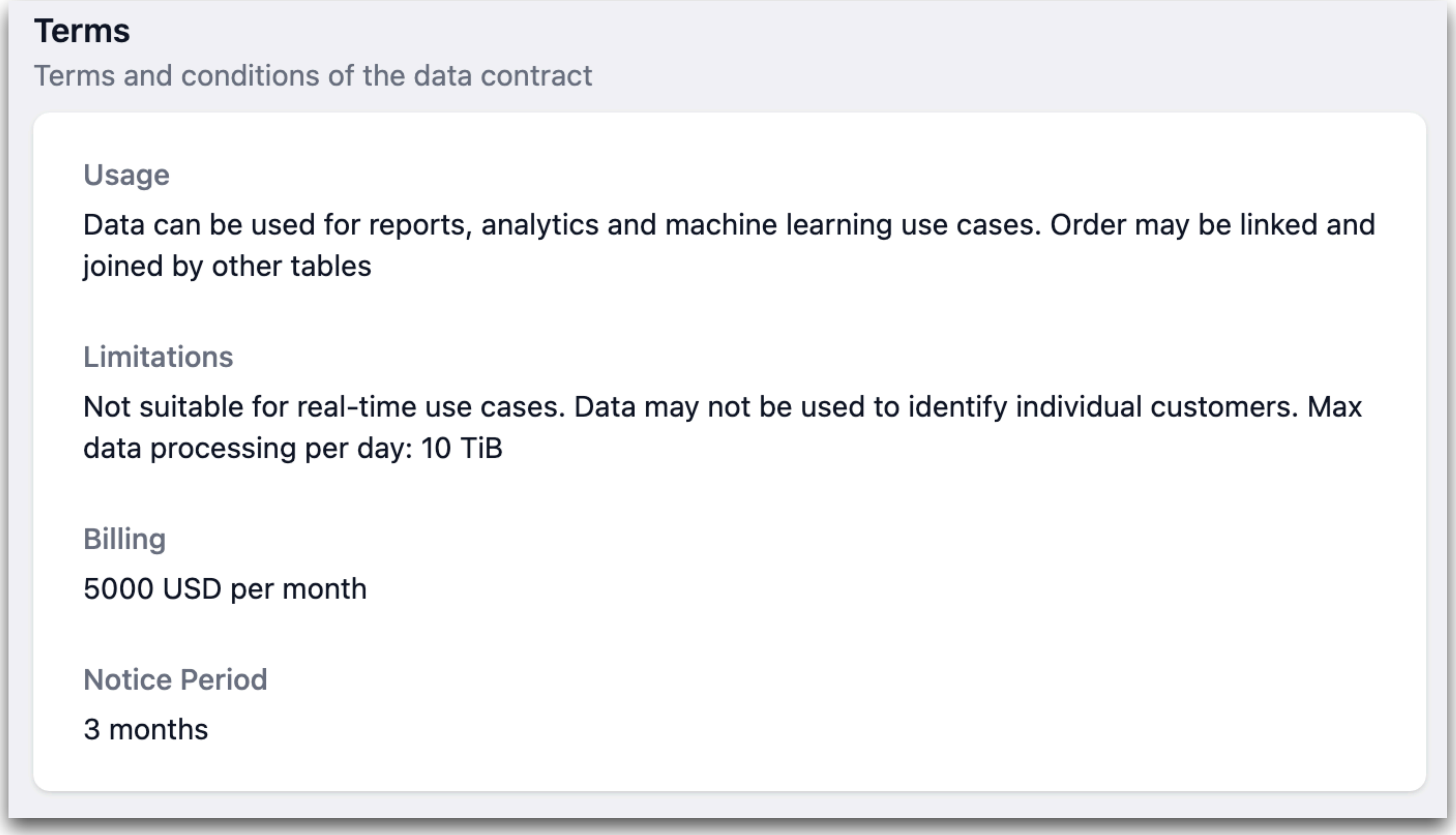
John Doe (Data Product Owner) john.doe@example.com



Servers

Servers of the data contract

Server	Type	Project	Dataset
production	BigQuery	acme_orders_prod	bigquery_orders_latest_npii_v1



Terms

Terms and conditions of the data contract

Usage

Data can be used for reports, analytics and machine learning use cases. Order may be linked and joined by other tables

Limitations

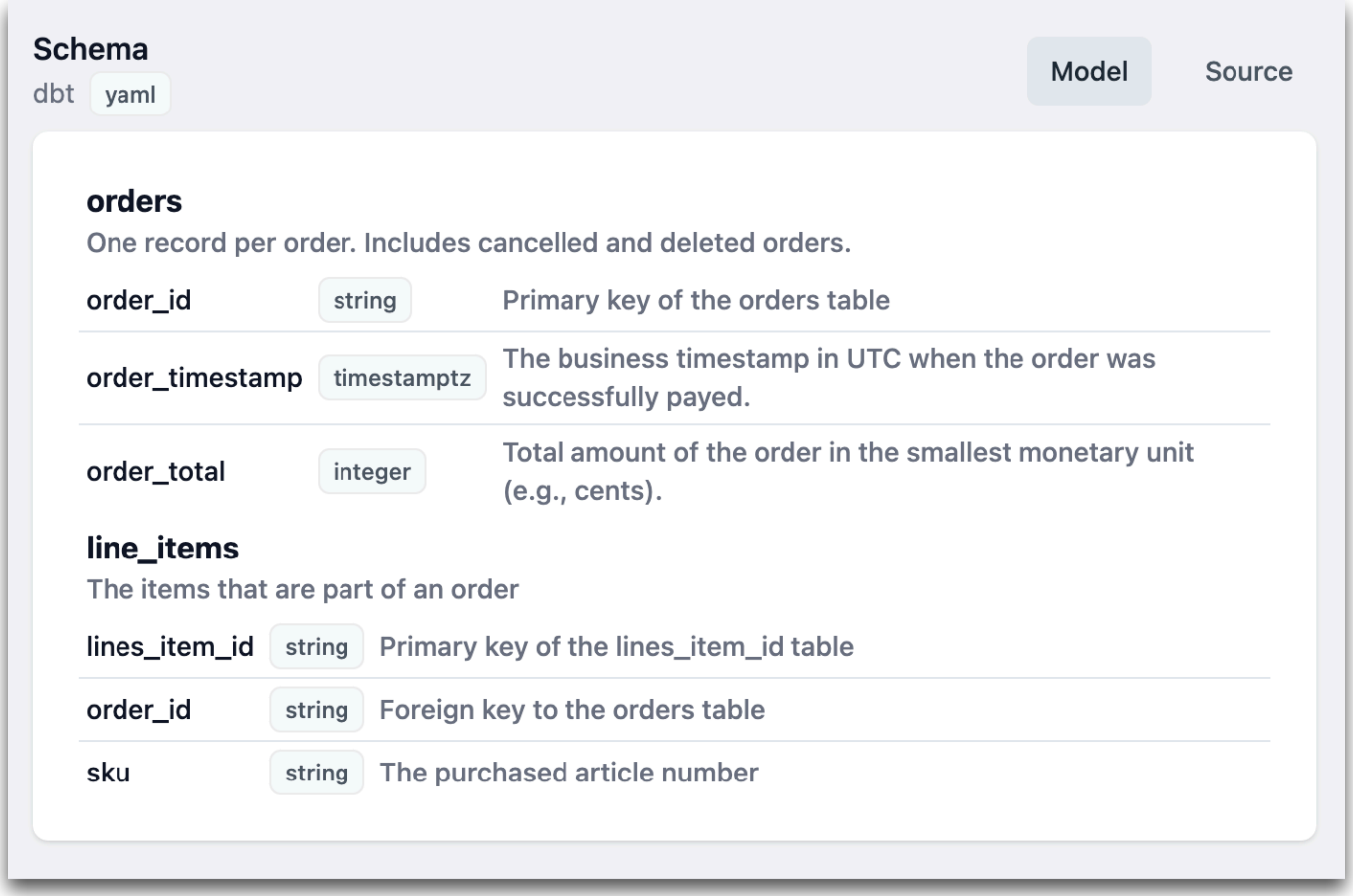
Not suitable for real-time use cases. Data may not be used to identify individual customers. Max data processing per day: 10 TiB

Billing

5000 USD per month

Notice Period

3 months



Schema

dbt yaml

orders

One record per order. Includes cancelled and deleted orders.

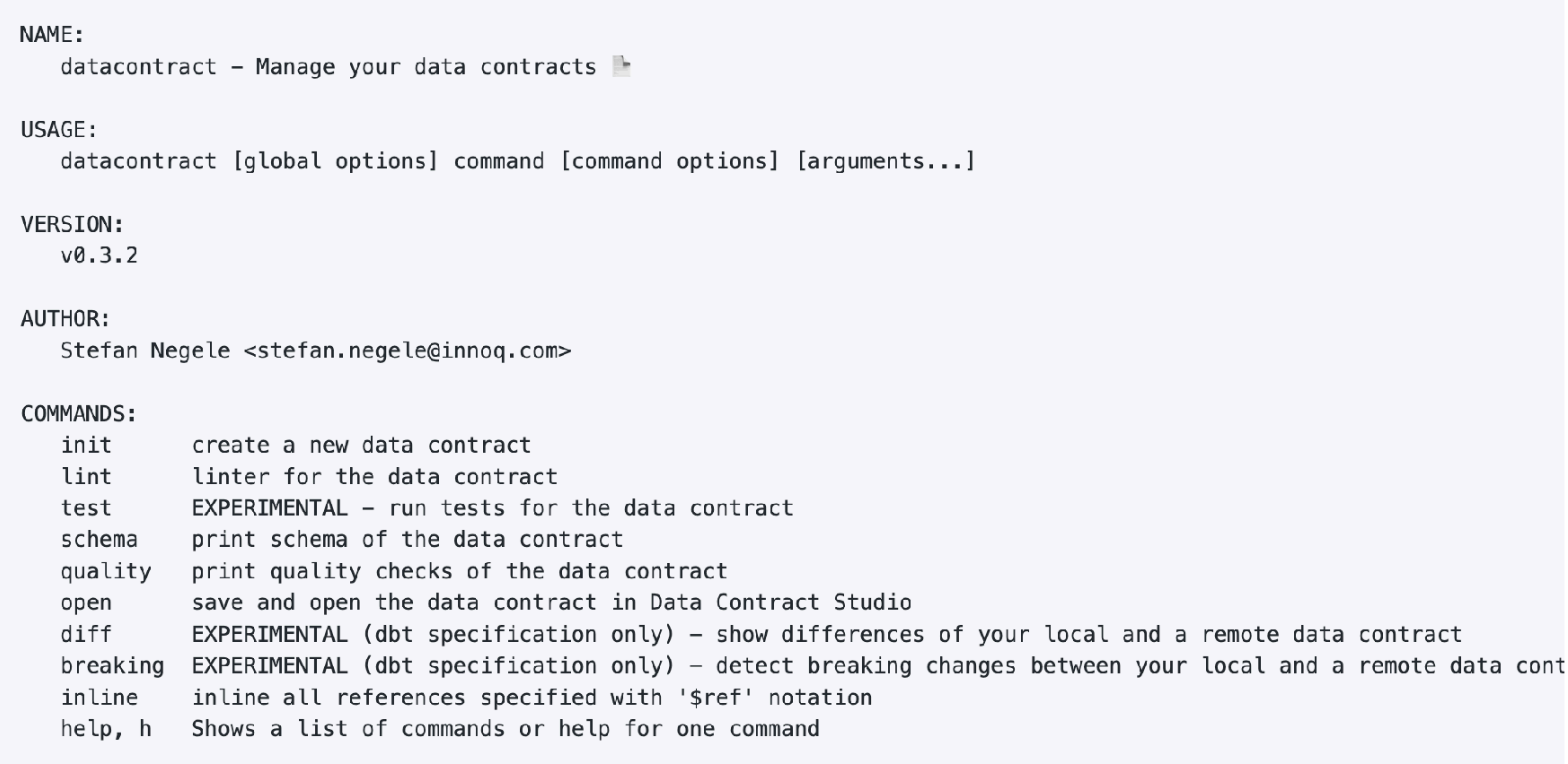
order_id	string	Primary key of the orders table
order_timestamp	timestamptz	The business timestamp in UTC when the order was successfully payed.
order_total	integer	Total amount of the order in the smallest monetary unit (e.g., cents).

line_items

The items that are part of an order

lines_item_id	string	Primary key of the lines_item_id table
order_id	string	Foreign key to the orders table
sku	string	The purchased article number

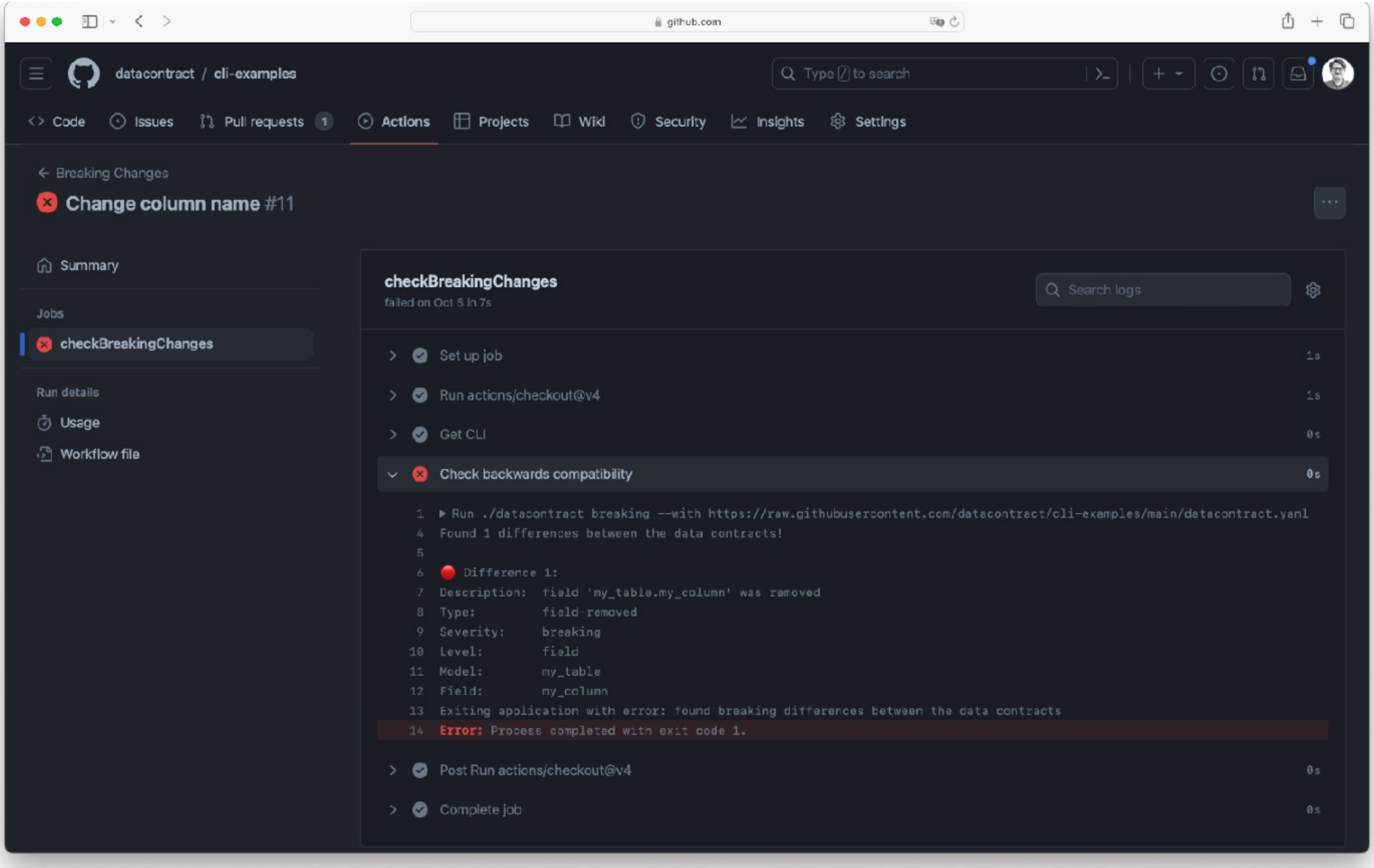
Can we automate anything other than documentation?



```
NAME:
   datacontract - Manage your data contracts 📄

USAGE:
   datacontract [global options] command [command options] [arguments...]

VERSION:
   v0.3.2

AUTHOR:
   Stefan Negele <stefan.negele@innoq.com>

COMMANDS:
   init       create a new data contract
   lint       linter for the data contract
   test       EXPERIMENTAL - run tests for the data contract
   schema     print schema of the data contract
   quality    print quality checks of the data contract
   open       save and open the data contract in Data Contract Studio
   diff       EXPERIMENTAL (dbt specification only) - show differences of your local and a remote data contract
   breaking   EXPERIMENTAL (dbt specification only) - detect breaking changes between your local and a remote data cont
   inline     inline all references specified with '$ref' notation
   help, h    Shows a list of commands or help for one command
```

cli.datacontract.com

```
~/demo
```



```
▶ Run ./datacontract breaking --with https://raw.githubusercontent.com/datacontract/cli-examples/main/datacontract.yaml
Found 1 differences between the data contracts!

🔴 Difference 1:
Description:  field 'my_table.my_column' was removed
Type:         field-removed
Severity:     breaking
Level:        field
Model:        my_table
Field:        my_column
Exiting application with error: found breaking differences between the data contracts
Error: Process completed with exit code 1.
```

```
$ datacontract test --test-options "-d duckdb_local -c quality/soda-conf.yml" --file contracts/data-contract
[...]
Creating quality directory if needed...
[18:01:15] Soda Core 3.0.51
[18:01:16] Scan summary:
[18:01:16] 2/2 checks PASSED:
[18:01:16]     transport_routes in duckdb_local
[18:01:16]       row_count between 90000 and 100000 [PASSED]
[18:01:16]       invalid_percent(freq) = 0 % [PASSED]
[18:01:16] All is good. No failures. No warnings. No errors.
```

Danke! Fragen?

INNOQ
www.innoq.com

Stefan Negele
stefan.negele@innoq.com

innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim
+49 2173 3366-0

Ohlauer Str. 43
10999 Berlin

Ludwigstr. 180E
63067 Offenbach

Kreuzstr. 16
80331 München

Wendenstraße 130
20537 Hamburg

Spichernstraße 44
50672 Köln

Königstorgraben 11
90402 Nürnberg